

Docket No.: P2002,0648

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313 20231.

By:  Date: September 17, 2003

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Dirk Fuhrmann et al.
Appl. No. : 10/633,996
Filed : August 4, 2003
Title : Integrated Memory and Method for Checking the Functioning of
an Integrated Memory

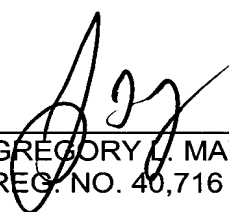
CLAIM FOR PRIORITY

Hon. Commissioner for Patents,
Alexandria, VA 22313-1450
Sir:

Claim is hereby made for a right of priority under Title 35, U.S. Code, Section 119, based upon the German Patent Application 102 35 454.5 filed August 2, 2002.

A certified copy of the above-mentioned foreign patent application is being submitted herewith.

Respectfully submitted,



GREGORY J. MAYBACK
REG. NO. 40,716

Date: September 17, 2003

Lerner and Greenberg, P.A.
Post Office Box 2480
Hollywood, FL 33022-2480
Tel: (954) 925-1100
Fax: (954) 925-1101

/mjb

BUNDESREPUBLIK DEUTSCHLAND



Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 102 35 454.5

Anmeldetag: 02. August 2002

Anmelder/Inhaber: Infineon Technologies AG, München/DE

Bezeichnung: Integrierter Speicher und Verfahren zur Funktions-
überprüfung eines integrierten Speichers

IPC: G 11 C 29/00

**Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-
sprünglichen Unterlagen dieser Patentanmeldung.**

München, den 9. Juli 2003
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

A handwritten signature in black ink, appearing to read 'Fausst'.

Fausst

Beschreibung

Integrierter Speicher und Verfahren zur Funktionsüberprüfung eines integrierten Speichers

5

Die vorliegende Erfindung betrifft einen integrierten Speicher sowie ein Verfahren zur Funktionsüberprüfung eines derartigen integrierten Speichers.

10 Integrierte Speicher, beispielsweise in Form von DRAMs (Dynamic Random Access Memories) werden im Herstellungsprozeß im allgemeinen umfangreichen Funktionstests unterzogen. Unter anderem dienen diese Funktionstests dazu, fehlerhafte Speicherzellen beziehungsweise fehlerhafte Spaltenleitungen oder
15 Reihenleitungen zu identifizieren. Mit zunehmender Speichergröße nehmen die Kosten von Funktionstests einen immer größeren Anteil an den gesamten Produktionskosten eines Speichers ein. Um die Testkosten zu senken, werden daher zunehmend Methoden wie Testmodes zur Komprimierung von Daten, Parallelisierung von Speicherzugriffen mittels einer reduzierten Anzahl von Anschlußkontakten oder zusätzliche Testlogik, beispielsweise in Form einer BIST-Einheit (Built-In Self-Test),
20 entwickelt.

25 Mit fortschreitender Entwicklung auf dem Gebiet von integrierten Speichern steigt die Betriebsfrequenz, mit der ein integrierter Speicher betrieben wird, im allgemeinen ständig an. Mit zunehmenden Betriebsfrequenzen von integrierten Speichern wird es meist schwieriger, diese auf ihre Funktionsfähigkeit hin zu testen. Dabei ist es für den Erhalt eines
30 weitgehend aussagekräftigen Testergebnisses wichtig, daß der Speicher auch bei seiner Betriebsfrequenz, die er im Normalbetrieb aufweist, getestet wird.

35 Erfahrungsgemäß ist es jedoch ein vergleichsweise großes Problem, Testgeräte für neuere integrierte Speicher bereitzustellen, die Ausgangssignale eines getesteten integrierten

Speichers, der mit maximaler Betriebsfrequenz getestet und betrieben wird, bei dieser geforderten Frequenz mit genügender Genauigkeit bewerten können. Oftmals sind derartige Testgeräte auf dem Markt nicht verfügbar oder vergleichsweise
5 teuer. Aus Kostengründen ist es deshalb oft von großem Vorteil, durch chipseitige Hardware Testgeräte älterer Bauart, die lediglich vergleichsweise niedrige Frequenzen unterstützen, für neuere Chip-Generationen nutzbar zu machen.

- 10 In diesem Zusammenhang sind Testmethoden bekannt, bei denen Testgeräte, die nur vergleichsweise niedrige Frequenzen unterstützen, nur noch zur Abfrage einer sogenannten Pass-Fail-Information eines Funktionstests eines Speicherzellenfeldes
15 Funktionstest selbst durch eine BIST-Einheit durchgeführt wird. Der Funktionstest mittels der BIST-Einheit kann unabhängig von der Betriebsfrequenz eines Testers mit der spezifizierten Zielfrequenz durchgeführt werden. Nachträglich wird das hochfrequenzbetriebene Speicherchip-Interface separat ge-
20 testet.

BIST-Einheiten sind, wenn diese eine möglichst flexible Adressgenerationseinheit besitzen sollen, nur schaltungs- und somit platzaufwendig realisierbar. Daher ist es in diesem Zu-
25 sammenhang interessanter, die Adressgenerierung durch ein extern angeschlossenes und flexibel programmierbares Testgerät zu steuern.

- Bisher wurde vor allem das Ziel verfolgt, Testpattern mit
30 möglichst einfacher inkrementeller beziehungsweise dekrementeller Adressgenerierung so zu entwerfen, daß eine möglichst einfache Abtestbarkeit des Speichers inklusive der Redundanz ohne komplexe Adressraum-Berechnungen möglich ist. Derartige Funktionstests werden vor allem auf Wafer-Ebene durchgeführt.
35 Bei der Generierung von adressierten Speicherzugriffen nimmt man aus Kostengründen hierbei jedoch folgende Einschränkungen vor: Es wird nur eine fix definierte Adress-Schrittweite, nur

eine Richtung der Inkrementierung sowie eine Adressierung entweder in Reihen- oder Spalten-Richtung unterstützt. Weiterhin werden keine komplexen Sprungmöglichkeiten zwischen dem redundanten Speicherzellenbereich und dem Hauptzellenfeld
5 unterstützt. Das Speicherchip-Interface ist meist nur einmalig vor Testbeginn konfigurierbar und während des Speichertests nicht ohne großen Zeitverlust umkonfigurierbar.

Der vorliegenden Erfindung liegt die Aufgabe zugrunde, einen
10 integrierten Speicher bereitzustellen, bei dem eine vergleichsweise hohe Funktionalität und Flexibilität bei der Adreßgeneration im Speichertest ermöglicht wird.

Weiterhin ist es Aufgabe der vorliegenden Erfindung, ein geeignetes Verfahren zur Funktionsüberprüfung eines derartigen
15 integrierten Speichers anzugeben.

Diese Aufgabe wird erfindungsgemäß durch einen integrierten Speicher gemäß Patentanspruch 1 und durch ein Verfahren zur
20 Funktionsüberprüfung eines derartigen integrierten Speichers gemäß Patentanspruch 12 gelöst.

Der Kern der vorliegenden Erfindung ist eine Adressierungs-
25 rechnungslogik des integrierten Speichers, die mit einer Adressierungseinheit zur Adressierung von Speicherzellen für einen Speicherzugriff anhand von empfangenen Adressierungssignalen verbunden ist. Diese Adressierungsberechnungslogik ist durch ein Testmodussignal für einen Testbetrieb des Speichers aktivierbar. Die Adressierungsberechnungslogik empfängt
30 Kommandosignale und Adreßsignale für den Testbetrieb, berechnet daraus für den Speicherzugriff die Adressierungssignale und speist diese in die Adressierungseinheit ein. Die Adressierungseinheit führt die eigentliche Adressierung von Speicherzellen für den Speicherzugriff während des Testbetriebs
35 durch. Die Adressierungsberechnungslogik kann ohne weiteres in eine komplexe BIST-Einheit integriert werden.

Gemäß dem erfindungsgemäßen Verfahren zur Funktionsüberprüfung des integrierten Speichers wird die Adressierungsberechnungslogik für den Testbetrieb durch das Testmodussignal aktiviert. Anschließend wird eine Initialisierung mit einer Übertragung von zu speichernden Ausgangsparametern in die Adressierungsberechnungslogik durchgeführt. Nach der Initialisierung werden die Kommandosignale und Adreßsignale für den Testbetrieb an die Adressierungsberechnungslogik angelegt und entsprechende Schreib-/Leseoperationen von der Zugriffssteuerung durchgeführt.

Mit der erfindungsgemäßen Adressierungsberechnungslogik ist die Voraussetzung geschaffen, daß zusätzliche Kommandos zur Testdurchführung von einem externen Testgerät bereitgestellt werden können, ohne daß hierzu der Hardware-Aufwand wesentlich zunimmt. Die Adressierungsberechnungslogik ist dazu geeignet, daß variable Schrittweiten und Sprungziele bei der Testdurchführung festgelegt werden können, wodurch eine erhöhte Funktionalität und Flexibilität beim Speichertest ermöglicht ist. Hierbei können jeweils geeignete Schrittweiten und Sprungziele bei der Initialisierung mit der Übertragung von entsprechenden Ausgangsparametern definiert werden. Mit dem Anlegen von zusätzlichen Kommandos zur Testdurchführung der Speichertest flexibel gestaltet werden.

Vorzugsweise ist die Adressierungsberechnungslogik derart ausgelegt, daß die Berechnung der Adressierungs-Signale zur Einspeisung in die Adressierungseinheit in der Adressierungsberechnungslogik innerhalb eines Taktzyklus eines an den Speicher angelegten Taktsignals erfolgt. Hierdurch wird eine schnelle Adreßberechnung gewährleistet (sogenannte On-the-fly-Adressberechnung). Eine derartige hochfrequenztaugliche Adreßberechnung kann nicht nur auf Wafer-Ebene, sondern auch auf Komponenten-Ebene insbesondere zur Verifikation der Fehlerabdeckung eines auf Wafer-Ebene durchgeführten Funktionstests (sogenannter Front-End-Test) verwendet werden.

Mit der Erfindung ist weiterhin vorteilhaft eine kombinierte oder getrennte Adressierung von regulärem Speicherzellenfeld und redundantem Speicherzellenfeld ermöglicht. Insbesondere wird zur getrennten Adressierung von redundantem Speicherzellenfeld mit der Initialisierung eine Adresse innerhalb des Redundanz-Speicherzellenfeldes als Zugriffs-Startadresse an die Adressierungsberechnungslogik übergeben.

Weitere vorteilhafte Aus- und Weiterbildungen der Erfindung sind in Unteransprüchen angegeben.

Die Erfindung wird im folgenden anhand der in der Zeichnung dargestellten Figuren, die Ausführungsbeispiele der vorliegenden Erfindung darstellen, näher erläutert. Es zeigen:

Figur 1 eine Ausführungsform eines erfindungsgemäßen integrierten Speichers,

Figur 2 eine Ausführungsform einer Adressierungseinheit und einer Adressierungsberechnungslogik gemäß der Erfindung,

Figur 3 eine detailliertere Ausführungsform einer Adressierungsberechnungslogik gemäß der Erfindung,

Figur 4 eine schematische Darstellung eines Speicherzellenfeldes des erfindungsgemäßen Speichers, das einem Funktionstest unterzogen wird,

Figur 5 schematische Darstellungen des Speicherzellenfeldes nach Figur 4 für unterschiedliche Testsequenzen.

In Figur 1 ist eine Ausführungsform eines erfindungsgemäßen integrierten Speichers 1 anhand eines prinzipiellen Aufbau-schemas gezeigt. Zur Ansteuerung des in Figur 1 dargestellten SDRAMs sind Anschlüsse an dem Baustein vorgesehen, die in

drei Gruppen aufgeteilt werden können. Es sind Adreßanschlüsse ADR, Kommandoanschlüsse CMD und Datenanschlüsse DQ vorgesehen. Während die Kommando- und Adreßanschlüsse CMD beziehungsweise ADR reine Eingangsanschlüsse sind, sind die Datenanschlüsse DQ Eingangs- und Ausgangsanschlüsse, je nachdem, ob eine Schreib- oder Lese-Operation durchgeführt wird. Die entsprechenden Signale werden von einer Pin-Logik 11 empfangen beziehungsweise ausgegeben.

10 Der SDRAM gemäß Figur 1 besteht im wesentlichen aus einer Zugriffssteuerung 6 und einem Speicherzellenfeld 4. Die Zugriffssteuerung 6 enthält mehrere Logik-Baugruppen zur Kommando-Interpretation und zur Adressierung. Insbesondere ist eine Adressierungseinheit 8, eine Testmoduslogik 9 sowie eine
15 Kommando-Logik 10 vorgesehen. Mit Hilfe dieser Baugruppen kann die Zugriffssteuerung 6 einen Speicherzugriff auf das Speicherzellenfeld 4 zum Lesen oder Schreiben von Daten steuern. Der Schreib- beziehungsweise Lese-Zugriff wird mit einem Aktivierungskommando des Speicherzellenfeldes zusammen mit
20 einer Reihenadressierung (X-Adressierung) eingeleitet, anschließend wird die zugehörige Spaltenadressierung (Y-Adressierung) zusammen mit dem Lese- oder Schreib-Kommando durchgeführt. In Figur 1 sowie in den folgenden Figuren 2 und 3 sind jeweilige Komponenten beziehungsweise Baugruppen zur
25 X-Adressierung und Y-Adressierung mit "X" beziehungsweise "Y" gekennzeichnet.

Für den Speicherzugriff werden entsprechende Reihenadressen und Spaltenadressen von der Adressierungseinheit 8 zu einem
30 Reihendecoder 2 beziehungsweise Spaltendecoder 3 weitergeleitet, die die entsprechenden adressierten Speicherzellen MC des Speicherzellenfeldes 4 und/oder Redundanz-Speicherzellenfeldes 5 auswählen. Die Speicherzellen MC sind hierbei in Kreuzungspunkten von Reihenleitungen RO und Spaltenleitungen
35 COL angeordnet. Der Speicherzugriff erfolgt in Synchronisation mit einem Taktsignal CK, das an der Zugriffssteuerung 6 anliegt. Der Speicher 1 weist außerdem eine Adressierungsbe-

rechnungslogik 7 auf, deren Aufbau und Funktionsweise in Zusammenhang mit den Figuren 2 bis 5 näher erläutert wird.

In Figur 2 ist eine Ausführungsform einer Adressierungseinheit 8 und einer Adressierungsberechnungslogik 7 dargestellt. Die Adressierungseinheit 8 enthält Standardkomponenten, die der Adressierungs-Spezifikation des Speichers Rechnung tragen. Sie dient insbesondere zur Adressierung von Speicherzellen für den Speicherzugriff anhand von empfangenen Adressierungssignalen XA und YA. Mit der Adressierungseinheit 8 ist die Adressierungsberechnungslogik 7 verbunden. Diese ist mittels einer Testmoduslogik 25 durch ein Testmodussignal TM für einen Testbetrieb des Speichers aktivierbar. Die Testmoduslogik 25 ist vorzugsweise an den Adreßanschlüssen ADR mit den Signalleitungen A00 bis A11 verbunden. Im allgemeinen wird die Anzahl der Adreßanschlüsse über diese Anzahl hinausgehen, was vor allem von der Speicherdichte abhängt. Um möglichst viel zusätzliche Funktionalität kodieren zu können, werden vorzugsweise alle verfügbaren Adreß-, Kommando- und/oder Datenpins umkodiert.

Über diese Signalleitungen werden im Testbetrieb kombinierte Kommandosignale und Adreßsignale für den Testbetrieb empfangen und an die Adressierungsberechnungslogik 7 weitergeleitet. Die Testmoduslogik 25 befindet sich hierbei im Zustand "On". Die Adressierungsberechnungslogik 7 empfängt die an den Signalleitungen A00 bis A11 anliegenden kombinierten Kommandosignale und Adreßsignale, berechnet daraus für den Speicherzugriff die Adressierungssignale XA und YA und speist diese in die Adressierungseinheit 8 ein. Diese steuert in entsprechender Weise den Reihendecoder 2 und Spaltendecoder 3 an. Hierbei ist in der Adressierungseinheit 8 im Y-Adressierungspfad eine zusätzliche Burst-Logik 27 vorgesehen.

Im Normalbetrieb ist die Testmoduslogik 25, gesteuert durch das Testmodussignal TM, im Zustand "Off". In diesem Fall wird die Standard-Adreßlogik 26 der Adressierungseinheit 8, das

heißt also die Adressierungseinheit 8 direkt zur Durchführung eines Standard-Adressierungsvorgangs für einen Speicherzugriff im Normalbetrieb angesteuert. In diesem Modus sind die Signalleitungen A00 bis A11 zur Übertragung von reinen Adreßsignalen vorgesehen.

Die Adressierungsberechnungslogik 7 umfaßt eine Registereinheit 21 zum Speichern von Adreßparametern, die für die anschließende Berechnung der Adressierungssignale XA und YA benötigt werden. Der Registereinheit 21 ist eine Rechenkaskade 22 nachgeschaltet, die die eigentliche Berechnung der Adressierungssignale durchführt. Eine Vergleichs- und Rückschreibelogik 23 dient zum Rückschreiben von aktuellen Adressierungssignalen in die Registereinheit 21. Zur Steuerung des Berechnungsvorganges und zur Einspeisung von Initialisierungswerten in die Registereinheit 21 und in die Rechenkaskade 22 ist eine Steuerungseinheit 24 vorgesehen, die mit der Registereinheit 21, der Rechenkaskade 22 und der Vergleichs- und Rückschreibelogik 23 verbunden ist. Die Steuerungseinheit 24 empfängt die über die Testmoduslogik 25 empfangenen kombinierten Kommando- und Adreßsignale, die an den Adreßanschlüssen ADR anliegen. Die Adressierungsberechnungslogik 7 ist derart ausgelegt, daß die Berechnung der Adressierungssignale XA und YA in der Adressierungsberechnungslogik innerhalb eines Taktzyklus des Taktsignals CK erfolgt. Die Adressierungsberechnungslogik 7 kann sowohl zur Berechnung von Reihendressen als auch zur Berechnung von Spaltenadressen verwendet werden.

In Figur 3 ist eine detailliertere Ausführungsform einer Adressierungsberechnungslogik 7 dargestellt. Die Registereinheit 21 enthält ein Offset-Register 211 zur Speicherung einer Reihen-Zugriffs-Startadresse XOffset sowie ein Register 212 zur Speicherung einer aktuellen Zugriffsadresse Xakt. Weiterhin ist ein Backup-Register 213 zur Speicherung einer Adresse Xsave in Form einer Adreß-Marke vorgesehen. Entsprechende Register 214 bis 216 sind auch für die Spaltenadressierung vor-

handen. Die Register 211 und 212 beziehungsweise 214 und 215 sind über jeweilige Addierer und Zwischenregister mit einem Multiplexer 217 verbunden. Sowohl die Addierer als auch der Multiplexer 217 sind von der Steuerungseinheit 24 steuerbar.

5 Ebenso ist der Austausch der Adreßparameter der Backup-Register 213, 216 mit den Registern 212, 215 steuerbar. Die Register 211 bis 216 werden bei einer Initialisierung mit einem jeweiligen Ausgangsparameter gesetzt. Über den Multiplexer 217 werden jeweilige Adreßparameter für die Reihenadressierung und Spaltenadressierung an die Rechenkaskade 22 ausgegeben.

Die Rechenkaskade 22 umfaßt Register R1 und R2 zur Festlegung von Schrittweiten und Sprungzielen bei der Adreßberechnung.

15 In den Registern R1 und R2 wird dazu ein entsprechender Parameterwert gespeichert. Die Rechenkaskade 22 empfängt über das Eingangsregister 221 den von der Registereinheit 21 ausgegebenen Adreßparameterwert. Dieser wird zusammen mit dem Registerwert des Registers R1 einem Addierer/Subtrahierer 224 zugeführt. Ein Zwischenergebnis der Addition beziehungsweise Subtraktion wird in dem Register 225 zwischengespeichert. Das Zwischenergebnis wird zusammen mit dem Registerwert des Registers R2 einem Addierer/Subtrahierer 226 zugeführt, das Ergebnis dieser Addition beziehungsweise Subtraktion in dem

25 Ausgangsregister 227 abgelegt.

Auf diese Weise wird die von der Registereinheit 21 empfangene Adresse modifiziert. Weiterhin weist die Rechenkaskade 22 einen Bypass 222 auf, um die empfangene Adresse unverändert auszugeben. Die Rechenkaskade 22 gemäß dem vorliegenden Ausführungsbeispiel enthält zwei Register R1, R2 und Paralleladdierer/-subtrahierer 224, 226, wobei gemäß dem erfindungsge-
30 mäßen Konzept mindestens ein Register beziehungsweise Addierer/Subtrahierer vorzusehen ist. Die Anordnung von mehreren
35 Registern in der Rechenkaskade 22 hat den Vorteil, daß mit größerer Flexibilität unterschiedliche Schrittweiten und Sprungziele bei der Adreßberechnung realisiert werden können.

- Überentsprechende Steuersignale der Steuerungseinheit 24 kann entschieden werden, welche der Register R1, R2 zur Berechnung von modifizierten Adreßsignalen hinzugezogen werden beziehungsweise ob die Komponenten 224 und 226 als Addierer oder Subtrahierer fungieren. Mittels der Steuerungseinheit 24 ist weiterhin steuerbar, ob der im Eingangsregister 221 gespeicherte Adreßwert direkt an das Ausgangsregister 227 weitergeleitet wird.
- 10 Der Rechenkaskade 22 ist eine Vergleichslogik 23 nachgeschaltet, die eine Registerschaltung 233 aufweist zur Speicherung von vorgegebenen Maximalwerten XMax und YMax für die Reihendressierung beziehungsweise Spaltenadressierung. Weiterhin werden in der Registerschaltung 233 die Werte XOffset und
- 15 YOffset gespeichert. Der von der Rechenkaskade 22 ausgegebene Wert wird im Eingangsregister 231 gespeichert. Eine wahlweise von der Steuerungseinheit 24 gesteuerte Vergleichsschaltung 232 vergleicht die Werte XMax und YMax der konfigurierbaren Registerschaltung 233 mit dem Wert des Eingangsregister 231.
- 20 Abhängig von dem Vergleichsergebnis wird die Adresse des Eingangsregisters 231 modifiziert. Im Falle eines Adreßbereichs der Größe einer ganzzahligen Zweierpotenz n kann die Adreßbereichseingrenzung von 0 bis 2^n-1 durch einfache UND-Operation mit einer Bereichsmaske Xmask oder Ymask erfolgen. Eine flexible, geradezu ideale "Adreßbereichsfilterung" (Einschränkung bei Bereichsüberlauf/-unterlauf) nicht geradzahlig oder unvollständig geradzahlig binärer Adreßbreite wird
- 25 durch eine Vergleichslogik 23 erreicht.
- 30 Der gegebenenfalls modifizierte Adreßwert wird im Ausgangsregister 234 abgelegt und in den jeweiligen X-Adressierungspfad beziehungsweise Y-Adressierungspfad zur Weiterleitung an die Adressierungseinheit 8 eingespeist. Über Rückkopplungsregister 235 und 236 werden die aktuellen Adressierungssignale in die dafür vorgesehenen Register 212, 215 der Registerereinheit 21 rückgeschrieben.
- 35

Die Zusammenschaltung von Registereinheit 21, Rechenkaskade 22 und Vergleichs- und Rückschreibelogik 23 in Verbindung mit der Rückkopplungsschleife ist so konstituiert, daß alle Paralleladditionen in mehreren Stufen zusammen mit den Vergleichsberechnungen innerhalb eines Taktes abgearbeitet und für die im nächsten Takt folgende Adreßberechnung verwendet werden können. Die Adreßberechnung für die X-Adressierung und Y-Adressierung wird seriell gemäß der Adressierungsreihenfolge zum Zugriff auf das Speicherzellenfeld des Speichers, bei dem zuerst die Reihen und anschließend die Spalten adressiert werden, durchgeführt.

Zu Beginn des Betriebs der Adressierungsberechnungslogik 7 muß eine Initialisierung derselben mit geeigneten Ausgangsparametern vorgenommen werden. Die Initialisierung beinhaltet das Konfigurieren der Registereinheit, das Konfigurieren der zur Berechnung hinzuziehbaren Register in der Rechenkaskade und das Laden von Registerwerten in der Vergleichs- und Rückschreibelogik.

Im folgenden wird die Funktionsweise der Adressierungsberechnungslogik 7 näher beschrieben.

Die Adressierungsberechnungslogik wird durch Aufruf eines Testmodes mittels des Testmodussignals TM einmalig aktiviert. Dieser Testmode bleibt bis zum Trennen der Stromversorgung aktiv oder wird durch einen entsprechenden Adreßberechnungs-Steuermodus aufgehoben. Die Adressierungsberechnungslogik ist vorteilhaft derart aufgebaut, daß ein erneutes Laden oder Umkonfigurieren von Registern nicht nötig ist.

Eine Initialisierung der Adressierungsberechnungslogik erfolgt entweder durch parallele, parallel-serielle oder rein serielle Übertragung von zu speichernden Registerwerten über die Adreßpins. Dabei können in einem oder mehreren nachfolgenden Takten die Register reihenweise gesetzt werden. Bevorzugt wird eine 2-Takt-Übertragung gewählt, bei der im ersten

- Takt das zu beschreibende Register adressiert wird und im zweiten oder nachfolgenden Takt die jeweiligen Parameterwerte parallel oder seriell übertragen werden. Bei einem 10 Bit breiten Adreßbus sind bei einer 2-Wort-Übertragung 1024 Register selektierbar und konfigurierbar. Alle in der Adressierungsberechnungslogik enthaltenen Register sind in ihrer Adreßbreite den zu adressierenden Speicherzellen angepaßt. Hierbei ist ein zusätzlicher Registersatz zur Adressierung einer Speicherbank (sogenannte Z-Adressierung) denkbar. Vorteilhaft ist ein spezielles Reset-Kommando vorgesehen, das alle Register in einem Takt auf entsprechend gewünschte Werte zurücksetzen kann. Somit wäre der Initialisierungsvorgang merklich verkürzbar.
- 15 Nach dem Konfigurieren der Register können von extern im Testbetrieb Schreib-/Lese-Operationen über die Kommandoanschlüsse eingeleitet werden. Im Testmodus werden an den Adreßanschlüssen, entgegen einer Betriebsweise im Normalbetrieb, kombinierte Kommando- und Adreßsignale angelegt, aus denen die Adressierungsberechnungslogik Testmodus-spezifische Adressen berechnet.

25 In der folgenden Tabelle sind beispielhafte, nicht abschließende Kommandos aufgelistet, die für einen Testbetrieb zur Verfügung stehen. Es werden hierbei nur X-Adreßbefehle aufgelistet, die analog auch für die Y-Adressierung beziehungsweise Z-Adressierung existieren.

| | Takte | Bedeutung |
|------------------|-------|---|
| Reset | 1 | Setze Adresslogik zurück |
| ExitTestMode | 1 | Deaktiviere Adresslogik |
| LoadReg n, m | 2 | Lade Register n mit dem Wert m |
| | | |
| AddXOffsetToXakt | 1 | Addiere den Offset zur aktuellen X-Addr |
| MoveXaktToRn | 1 | Kopiere Xakt nach Register Rn |
| MoveXaktToXsave | 1 | Kopiere Xakt nach Xsave |
| MoveXsaveToXakt | 1 | Kopiere Xsave nach Xakt |
| MoveXaktToXMax | 1 | Kopiere Xakt nach XMax |

| | | |
|----------|---|---|
| | | |
| XALUSubn | 1 | Subtrahiere alle Register Rn von Xakt je nach Bit-Status von Argument1. Argument1 gibt die Registerauswahl an. |
| XALUAddn | 1 | Addiere alle Register Rn zu Xakt je nach Bit-Status von Argument1. Argument1 gibt die Registerauswahl an. |
| XALUn,x | 1 | Addiere/Subtrahiere alle Register zu/von Xakt, je nach Bit-Status von Argument1 und Argument2. Argument1 gibt die Registerauswahl an. Argument 2 gibt die Addition (=0) bzw. Subtraktion (=1) bei der Registerauswahl an. |

Im folgenden ist eine beispielhafte Adreßcodierungstabelle für eine Adressierungsberechnungslogik mit drei Registern R1, R2, R3 in der Rechenkaskade und je einem Offset-, Max- und

5 Save-Register für die X-Adressierung und Y-Adressierung dargestellt.

| Befehl | Befehlskodierung | | | | | | Argument2 | | | Argument1 | | |
|-----------------|------------------|-----|-----|-----|-----|-----|-----------|-----|-----|-----------|-----|-----|
| | A11 | A10 | A09 | A08 | A07 | A06 | A05 | A04 | A03 | A02 | A01 | A00 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| ExitTestMode | 1 | 1 | 1 | 1 | 1 | 1 | | | | | | |
| LoadReg n, m | 0 | 0 | 0 | 0 | 0 | 1 | m2 | m1 | m0 | R3 | R2 | R1 |
| AddXOffToXakt | 0 | 0 | 0 | 0 | 0 | 1 | | | | | | |
| MoveXaktToRn | 0 | 0 | 0 | 0 | 1 | 0 | | | | R3 | R2 | R1 |
| MoveXaktToXSave | 0 | 0 | 0 | 0 | 1 | 1 | | | | | | |
| MoveXSaveToXakt | 0 | 0 | 0 | 1 | 0 | 0 | | | | | | |
| MoveXaktToXMax | 0 | 0 | 0 | 1 | 0 | 1 | | | | | | |
| AddYOffToXakt | 1 | 0 | 0 | 0 | 0 | 1 | | | | | | |
| MoveYaktToRn | 1 | 0 | 0 | 0 | 1 | 0 | | | | R3 | R2 | R1 |
| MoveYaktToXSave | 1 | 0 | 0 | 0 | 1 | 1 | | | | | | |
| MoveYSaveToXakt | 1 | 0 | 0 | 1 | 0 | 0 | | | | | | |
| MoveYaktToXMax | 1 | 0 | 0 | 1 | 0 | 1 | | | | | | |
| XALUSubn | 0 | 1 | 0 | 0 | 0 | 0 | | | | R3 | R2 | R1 |
| XALUAddn | 0 | 1 | 0 | 0 | 0 | 0 | | | | R3 | R2 | R1 |
| XALUn,x | 0 | 1 | 0 | 0 | 0 | 0 | x2 | x1 | x0 | R3 | R2 | R1 |
| YALUSubn | 1 | 0 | 0 | 0 | 0 | 0 | | | | R3 | R2 | R1 |
| YALUAddn | 1 | 0 | 0 | 0 | 0 | 0 | | | | R3 | R2 | R1 |
| YALUn,x | 1 | 0 | 0 | 0 | 0 | 0 | x2 | x1 | x0 | R3 | R2 | R1 |

Im folgenden wird beispielhaft die Durchführung der bestimmten Operationsfolge "ACT x+1 / WR Y+1" erläutert.

Dies bedeutet, daß Reihen nacheinander mit Schrittweite 1 adressiert und aktiviert werden und Spalten ebenfalls mit Schrittweite 1 nacheinander beschrieben werden. Nach der Initialisierung des Speichers, bei der im Register R1 der Wert 1 abgelegt worden ist, wird an die Kommandoanschlüsse und Adreßanschlüsse folgende Befehlsfolge angelegt:

ACT XALUAdd1 / WR YALUAdd1.

Im allgemeinen kann die Berechnungslogik aus der aktuellen X-Adresse Xakt, den Registerwerten der Register Rx, dem Offset-Wert XOffset und XMax die nächste gültige X-Adresse berechnen gemäß

$$X_{akt} = X_{akt} + \sum_{x=1,2,\dots} R_x$$

falls $X_{offset} < X_{akt} + \sum_{x=1,2,\dots} R_x < X_{max}$

$$X_{akt} = X_{akt} + \sum_{x=1,2,\dots} R_x - (X_{max} - X_{offset})$$

falls $X_{akt} + \sum_{x=1,2,\dots} R_x > X_{max}$

$$X_{akt} = X_{akt} + \sum_{x=1,2,\dots} R_x + (X_{max} - X_{offset})$$

falls $X_{akt} + \sum_{x=1,2,\dots} R_x < X_{offset}$

Für die Y-Adreßberechnung ergibt sich entsprechendes. Die obigen Berechnungsgleichungen erfordern bei einer einstufigen Rechenkaskade mindestens drei Berechnungen, wobei der Größer-Kleiner-Vergleich durch eine Addition/Subtraktion implizit schon vorliegt. Mit entsprechend gewählten X-Y-Offset-Registerwerten kann bestimmt werden, ob ein regulärer Zellen-

felddurchlauf mit Overflow-Automatik, ein Durchlauf eines Bereichs des Zellenfelds mit Overflow-Automatik, ein Durchlauf des Redundanz-Zellenfeldes mit Overflow-Automatik oder ein Durchlauf eines Bereichs des Redundanz-Zellenfeldes mit Overflow-Automatik durchgeführt wird. Hierfür werden die X/Y-Offset-Werte und X/YMax-Werte entsprechend gewählt. Der Registerwert Rx gibt hierbei die Schrittweite und Schrittrichtung an.

10 In den Figuren 4 und 5 ist eine schematische Darstellung eines Speicherzellenfeldes 4 beziehungsweise eines Redundanz-Speicherzellenfeldes 5 mit einer beispielhaften Anzahl an Reihen und Spalten gezeigt. Das Speicherzellenfeld 4 weist 32 Reihen und 64 Spalten auf. Es sind in dem Speicherzellenfeld 15 4 demzufolge 32 x 64 Speicherzellen angeordnet. Das X-Redundanz-Speicherzellenfeld hat die Größe 2 x 64, das Y-Redundanz-Speicherzellenfeld die Größe 32 x 4.

Im Zusammenhang mit Figur 5 werden im folgenden vier Beispiele eines Adressierungsdurchlaufs für einen Testbetrieb erläutert.

25 In Figur 5a ist ein Adressierungsdurchlauf des X-Y-Redundanz-Speicherzellenfeldes mit Schrittweite +1 gezeigt (schraffiert dargestelltes Zellenfeld). Hierzu werden für die Adressierungsberechnungslogik folgende Start-Parameter bei der Initialisierung gewählt:

XOffset=32
30 YOffset=64
Xakt=XOffset
Yakt=YOffset
XMax=33
YMax=67
35 Rl=+1

In Figur 5b ist ein Adressierungsdurchlauf des regulären Speicherzellenfeldes 4 mit Schrittweite +2 dargestellt. Hierbei werden folgende Start-Parameter gesetzt:

```
5  XOffset=0
   YOffset=0
   Xakt=XOffset
   Yakt=YOffset
   XMax=31
10  YMax=63
   R1=+2
```

In Figur 5c ist ein Adressierungsdurchlauf des X-Redundanz-Speicherzellenfeldes mit Schrittweite -1 gezeigt. Hierbei

15 werden folgende Start-Parameter gesetzt:

```
   XOffset=32
   YOffset=0
   Xakt=XOffset
20  Yakt=YOffset
   XMax=33
   YMax=67
   R1=-1
```

25 In Figur 5d ist ein Adressierungsdurchlauf des kompletten Speicherzellenfeldes mit Schrittweite -2 dargestellt. Hierbei werden folgende Start-Parameter gewählt:

```
   XOffset=0
30  YOffset=0
   Xakt=XOffset
   Yakt=YOffset
   XMax=33
   YMax=67
35  R1=-2
```

Patentansprüche

1. Integrierter Speicher

- mit einem Speicherzellenfeld (4, 5) zur Speicherung von Daten,
- mit einer Zugriffssteuerung (6) zur Steuerung eines Speicherzugriffs auf das Speicherzellenfeld zum Lesen oder Schreiben von Daten,
- mit einer Adressierungseinheit (8) zur Adressierung von Speicherzellen (MC) für den Speicherzugriff anhand von empfangenen Adressierungssignalen (XA, YA),
- mit einer Adressierungsberechnungslogik (7), die mit der Adressierungseinheit (8) verbunden ist,
- wobei die Adressierungsberechnungslogik (7) durch ein Testmodussignal (TM) für einen Testbetrieb des Speichers aktivierbar ist,
- wobei die Adressierungsberechnungslogik (7) Kommandosignale und Adreßsignale für den Testbetrieb empfängt, daraus für den Speicherzugriff die Adressierungssignale (XA, YA) berechnet und in die Adressierungseinheit (8) einspeist.

2. Integrierter Speicher nach Anspruch 1,

- d a d u r c h g e k e n n z e i c h n e t, daß
- die Adressierungsberechnungslogik (7) die Kommandosignale und Adreßsignale für den Testbetrieb über Signalleitungen (A00 bis A11) empfängt, die in einem Normalbetrieb des Speichers zum Empfang von Adreßsignalen, Kommandosignalen oder Datensignalen oder einer Kombination derer dienen.

3. Integrierter Speicher nach Anspruch 1 oder 2,

- d a d u r c h g e k e n n z e i c h n e t, daß
- der Speicherzugriff in Synchronisation mit einem Taktsignal (CK) erfolgt,
 - die Adressierungsberechnungslogik (7) derart ausgelegt ist, daß die Berechnung der Adressierungssignale (XA, YA) in der Adressierungsberechnungslogik innerhalb eines Taktzyklus erfolgt.

4. Integrierter Speicher nach einem der Ansprüche 1 bis 3,
d a d u r c h g e k e n n z e i c h n e t, daß
die Adressierungsberechnungslogik (7) wenigstens ein Register
5 (R1, R2) aufweist zur Festlegung von Schrittweiten und/oder
Sprungzielen bei der Adreßberechnung.

5. Integrierter Speicher nach einem der Ansprüche 1 bis 4,
d a d u r c h g e k e n n z e i c h n e t, daß
10 die Adressierungsberechnungslogik (7) umfaßt:
- eine Registereinheit (21) zum Speichern von Adreßparametern
für die Berechnung der Adressierungssignale,
- eine der Registereinheit (21) nachgeschaltete Rechenkaskade
(22) zur Berechnung der Adressierungssignale (XA, YA),
15 - eine der Rechenkaskade (22) nachgeschaltete Rückschreibelo-
gik (23) zum Rückschreiben von aktuellen Adressierungssigna-
len (Xakt) in die Registereinheit (21),
- eine Steuerungseinheit (24), die zur Steuerung des Berech-
nungsvorgangs und zur Einspeisung von Initialisierungswerten
20 mit der Registereinheit (21) und der Rechenkaskade (22) ver-
bunden ist.

6. Integrierter Speicher nach Anspruch 5,
d a d u r c h g e k e n n z e i c h n e t, daß
25 - die Registereinheit ein Offset-Register (211, 214) zur
Speicherung einer Zugriffs-Startadresse (XOffset, YOffset),
insbesondere einer Zugriffs-Startadresse innerhalb eines Red-
undanz-Speicherzellenfeldes (5) umfaßt,
- die Registereinheit ein Register (212, 215) zur Speicherung
30 einer aktuellen Zugriffsadresse (Xakt, Yakt) umfaßt und diese
an die Rechenkaskade (22) ausgibt.

7. Integrierter Speicher nach Anspruch 5 oder 6,
d a d u r c h g e k e n n z e i c h n e t, daß
35 die Rechenkaskade (22)
- eine Adresse der Registereinheit (21) empfängt,

- wenigstens ein Register (R1, R2) zur Speicherung eines Parameterwerts aufweist,
- wenigstens einen Addierer und/oder Subtrahierer (224, 226) aufweist zur Addition beziehungsweise Subtraktion des Parameterwerts mit beziehungsweise von der empfangenen Adresse zu einer modifizierten Adresse und diese ausgibt,
- einen Bypass (222) zum Addierer beziehungsweise Subtrahierer aufweist, um die empfangene Adresse unverändert auszugeben.

10

8. Integrierter Speicher nach Anspruch 7, dadurch gekennzeichnet, daß durch die Steuerungseinheit (24) gesteuert wird, ob die empfangene Adresse modifiziert wird und ob in diesem Fall eine Addition und/oder Subtraktion von einem oder mehreren Parameterwerten erfolgt.

15

9. Integrierter Speicher nach Anspruch 7 oder 8, dadurch gekennzeichnet, daß

20

- der Rechenkaskade (22) eine Vergleichslogik (23) nachgeschaltet ist zum Vergleich eines vorgegebenen Maximalwertes (Xmax, Ymax) einer Adresse mit der ausgegebenen Adresse der Rechenkaskade,

25

- wobei abhängig von einem Vergleichsergebnis die ausgegebene Adresse modifiziert in die Adressierungseinheit (8) für den Speicherzugriff eingespeist wird.

10. Integrierter Speicher nach einem der Ansprüche 1 bis 9, dadurch gekennzeichnet, daß

30

- das Speicherzellenfeld (4) Reihen (RO) und Spalten (COL) aufweist, in deren Kreuzungspunkten Speicherzellen (MC) angeordnet sind,

- die Adressierungsberechnungslogik (7) sowohl zur Berechnung von Reihenadressen (XA) als auch zur Berechnung von Spaltenadressen (YA) verwendbar ist.

35

11. Integrierter Speicher nach Anspruch 10,

d a d u r c h g e k e n n z e i c h n e t, daß

die Adressierungsberechnungslogik (7) umfaßt:

- eine Registereinheit (21) mit getrennten Registern (211 bis 216) jeweils zum Speichern von Reihen- und Spalten-Adreßparametern für die Berechnung von Reihen- und Spalten-Adressierungssignalen (XA, YA),
- eine der Registereinheit (21) über einen Multiplexer (217) nachgeschaltete Rechenkaskade (22) zur seriellen Berechnung der Reihen- und Spalten-Adressierungssignale (XA, YA),
- 10 - eine der Rechenkaskade (22) nachgeschaltete Rückschreibelogik (23) zum jeweiligen Rückschreiben von aktuellen Reihen- und Spalten-Adressierungssignalen (Xakt, Yakt) in die Registereinheit (21),
- eine Steuerungseinheit (24), die zur Steuerung des Berechnungsvorgangs und zur Einspeisung von Initialisierungswerten mit der Registereinheit (21) und der Rechenkaskade (22) verbunden ist.

12. Verfahren zur Funktionsüberprüfung eines integrierten Speichers nach einem der vorhergehenden Ansprüche,

- bei dem die Adressierungsberechnungslogik (7) für einen Testbetrieb durch das Testmodussignal (TM) aktiviert wird,
- bei dem eine Initialisierung mit einer Übertragung von zu speichernden Ausgangsparametern in die Adressierungsberechnungslogik (7) durchgeführt wird,
- 25 - bei dem nach der Initialisierung die Kommandosignale und Adreßsignale für den Testbetrieb an die Adressierungsberechnungslogik (7) angelegt und Schreib-/Leseoperationen von der Zugriffssteuerung (6) durchgeführt werden.

30

13. Verfahren nach Anspruch 12,

d a d u r c h g e k e n n z e i c h n e t, daß

- mit der Initialisierung eine Adresse innerhalb eines Redundanz-Speicherzellenfeldes (5) als Zugriffs-Startadresse an
- 35 die Adressierungsberechnungslogik (7) übergeben wird.

Zusammenfassung

Integrierter Speicher und Verfahren zur Funktionsüberprüfung eines integrierten Speichers

5

Ein integrierter Speicher enthält eine Adressierungseinheit (8) zur Adressierung von Speicherzellen (MC) für einen Speicherzugriff anhand von empfangenen Adressierungssignalen (XA, YA). Eine Adressierungsberechnungslogik (7) ist mit der

10 Adressierungseinheit (8) verbunden. Diese ist durch ein Testmodussignal (TM) für einen Testbetrieb des Speichers aktivierbar. Die Adressierungsberechnungslogik (7) empfängt Kommandosignale und Adreßsignale für den Testbetrieb, berechnet daraus die Adressierungssignale (XA, YA) für den Speicherzu-

15 griff und speist diese in die Adressierungseinheit (8) ein. Nach einer Initialisierung mit dem Laden von Ausgangsparametern werden die Kommandosignale und Adreßsignale für den Testbetrieb an die Adressierungsberechnungslogik (7) angelegt und Schreib-/Leseoperationen von einer Zugriffssteuerung (6)

20 durchgeführt. Erfindungsgemäß wird ein integrierter Speicher mit implementierter BIST-Hardware bereitgestellt, bei dem dennoch eine vergleichsweise hohe Funktionalität und Flexibilität beim Speichertest ermöglicht ist.

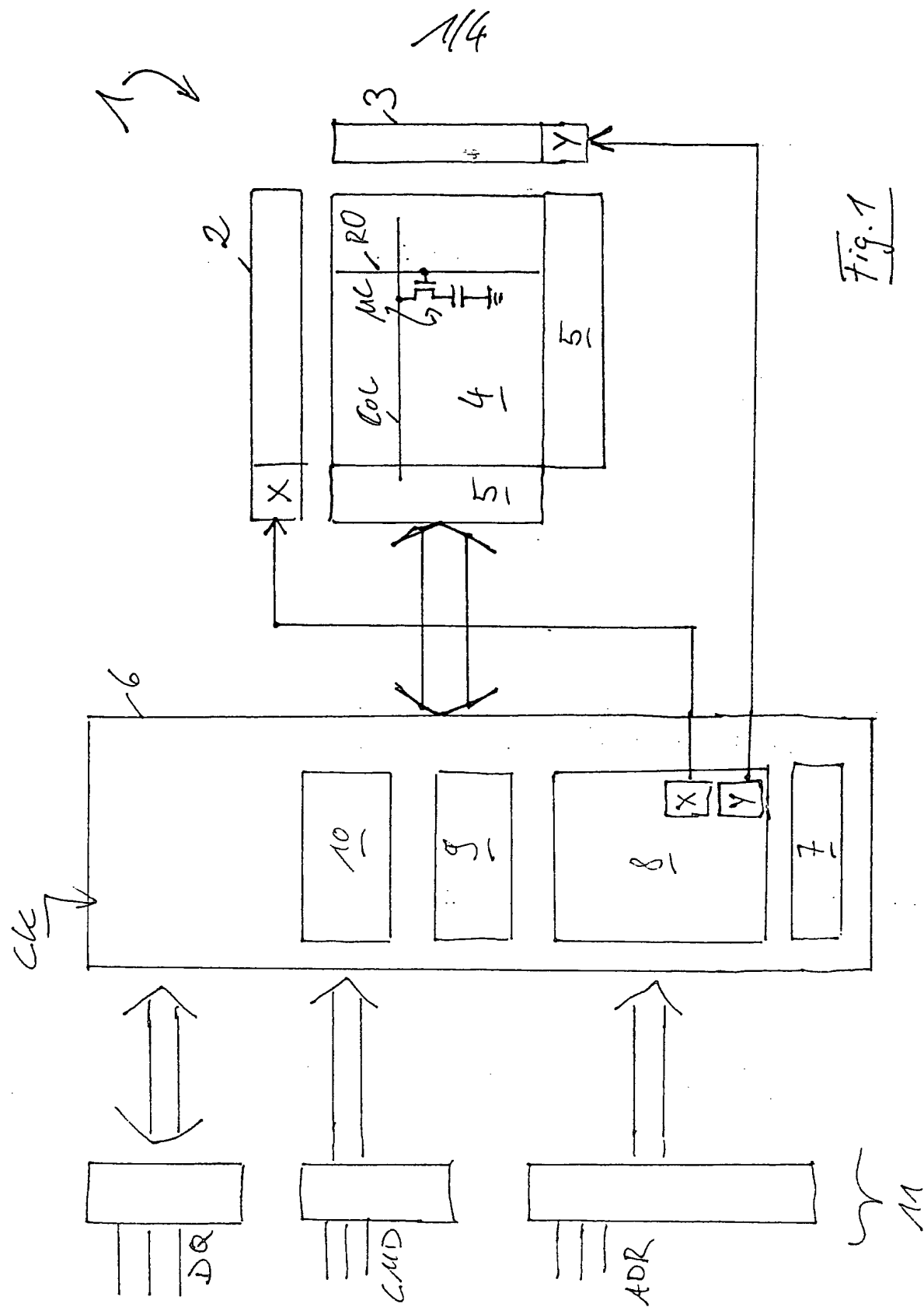
5

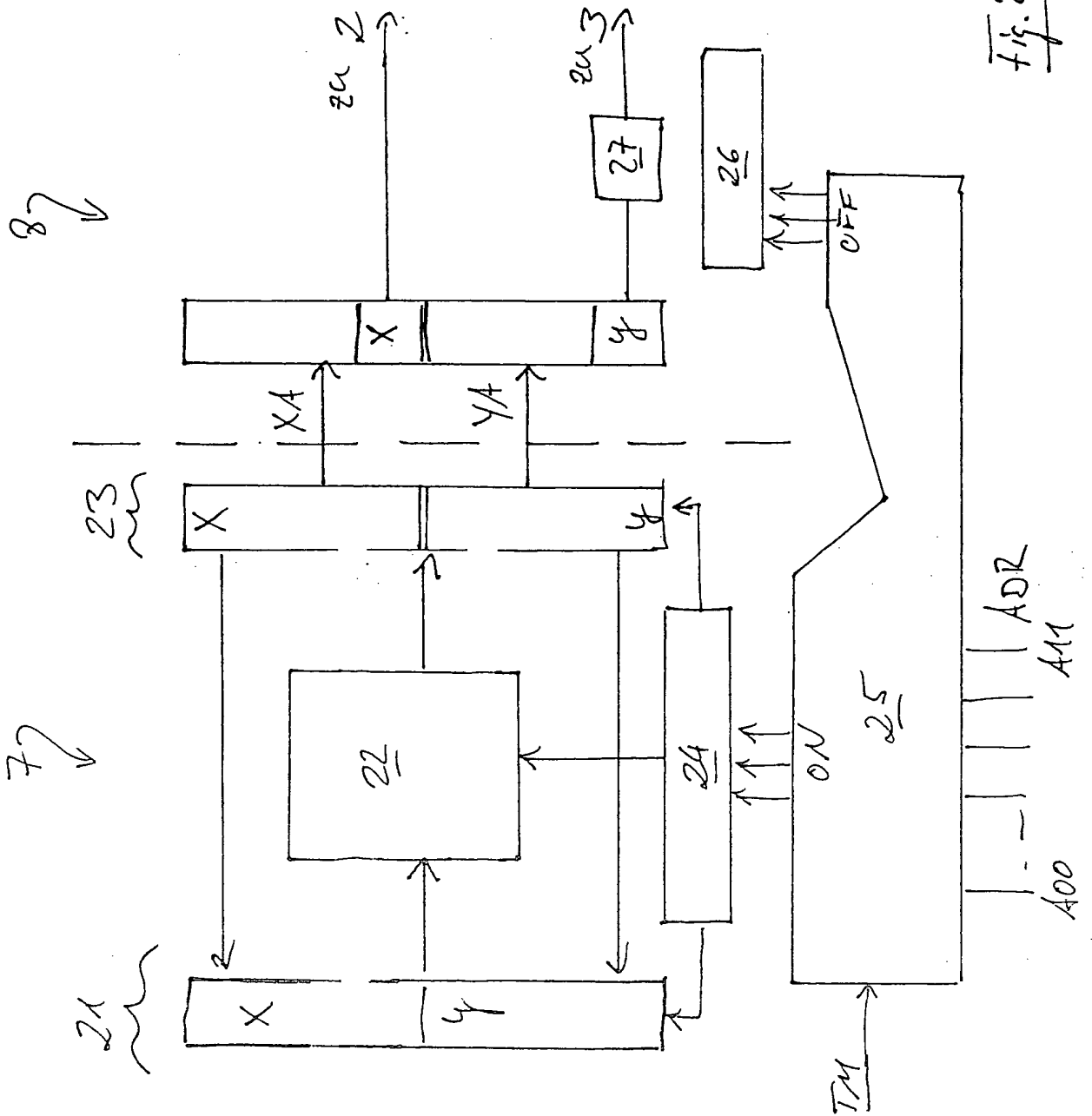
Figur 2

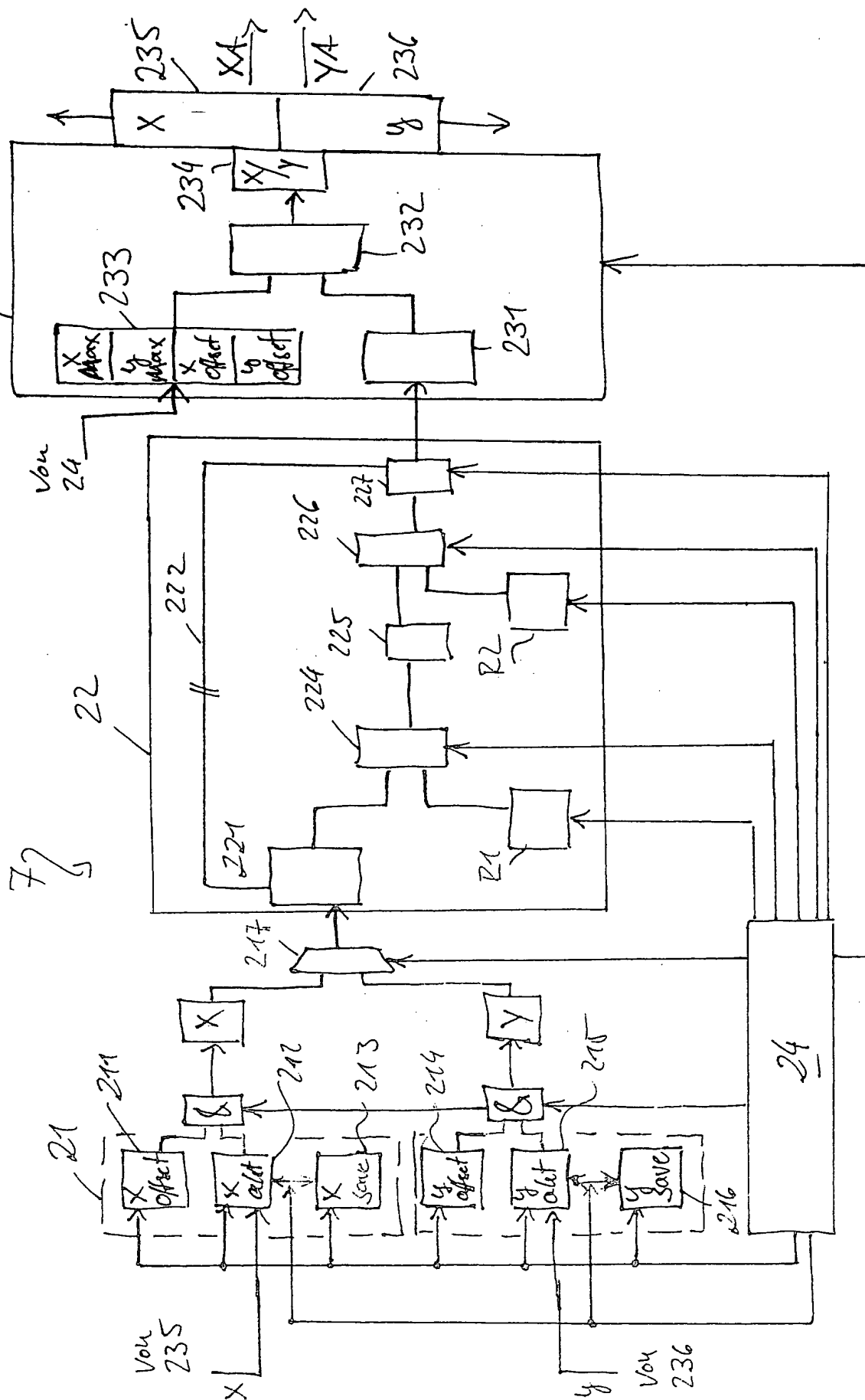
Bezugszeichenliste

| | | |
|----|-------------|-----------------------------------|
| | 1 | Speicher |
| | 2 | Reihendecoder |
| 5 | 3 | Spaltendecoder |
| | 4 | Speicherzellenfeld |
| | 5 | Redundanz-Speicherzellenfeld |
| | 6 | Zugriffssteuerung |
| | 7 | Adressierungsberechnungslogik |
| 10 | 8 | Adressierungseinheit |
| | 9 | Testmoduslogik |
| | 10 | Kommandologik |
| | 11 | Pin-Logik |
| | 21 | Registereinheit |
| 15 | 22 | Rechenkaskade |
| | 23 | Vergleichs- und Rückschreibelogik |
| | 24 | Steuerungseinheit |
| | 25 | Testmoduslogik |
| | 26 | Standard-Adreßlogik |
| 20 | 27 | Burst-Logik |
| | 211 bis 216 | Register |
| | 217 | Multiplexer |
| | 221 | Eingangsregister |
| | 222 | Bypass |
| 25 | 224 | Addierer/Subtrahierer |
| | 225 | Register |
| | 226 | Addierer/Subtrahierer |
| | 227 | Ausgangsregister |
| | 231 | Eingangsregister |
| 30 | 232 | Vergleichsschaltung |
| | 233 | Registerschaltung |
| | 234 | Ausgangsregister |
| | 235, 236 | Rückkopplungsregister |
| | MC | Speicherzellen |
| 35 | DQ | Datenanschlüsse |
| | CMD | Kommandoanschlüsse |
| | ADR | Adreßanschlüsse |

| | | |
|----|-------------|------------------------------------|
| | X | Komponente der Reihenadressierung |
| | Y | Komponente der Spaltenadressierung |
| | RO | Reihe |
| | COL | Spalte |
| 5 | TM | Testmodussignal |
| | R1,R2 | Register |
| | CK | Taktsignal |
| | XA | Adressierungssignal |
| | YA | Adressierungssignal |
| 10 | A00 bis A11 | Signalleitung |
| | XOffset | Reihen-Zugriffs-Startadresse |
| | Xakt | aktuelle Adresse |
| | Xsave | Backup-Adresse |
| | YOffset | Spalten-Zugriffs-Startadresse |
| 15 | Yakt | aktuelle Adresse |
| | Ysave | Backup-Adresse |







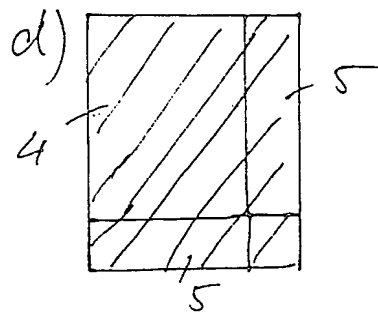
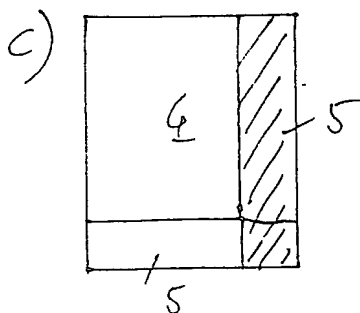
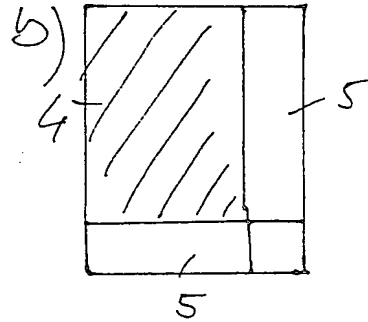
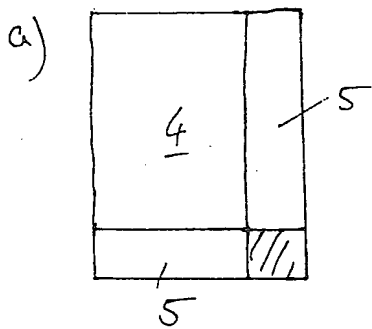
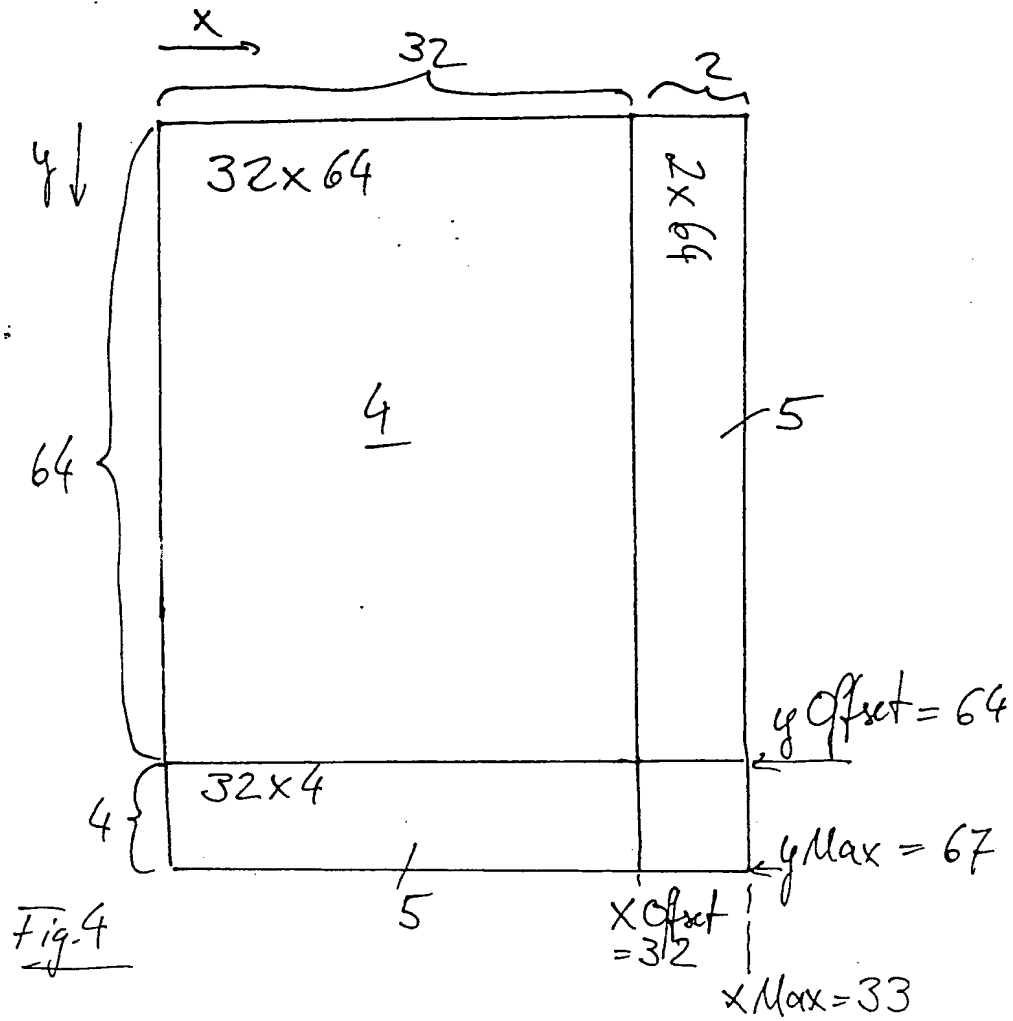


Fig. 5